Pebbling of Oriented Graphs

SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Jerad Scott DeVries

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

Dalibor Froncek

September, 2017

Abstract

In traditional graph pebbling a move across an edge is made by removing two pebbles from one vertex and adding one pebble to an adjacent vertex. We extend this concept to oriented graphs by subtracting three pebbles when moving against an edge orientation and two pebbles when moving with an edge orientation. The cover pebbling number of an oriented graph is the minimum number of pebbles such that given any initial placement of these pebbles we can simultaneously place a pebble on every vertex. In this paper we will look at pebblings of oriented paths.

Contents

Chapter 1

Introduction

Graph pebbling is a two player game played on a graph where pebbles are distributed on the vertices. It arose from an attempt by Lagarias and Saks to give an alternate proof to a theorem about zero-sum sequences (see introduction in [1]). This theorem, proven by Lemke and Kleitman [8] in 1989, states that for every sequence S of n elements from the group $\mathbb{Z}_n$ there exists a zero-sum subsequence T such that $\sum_{t \in T} \frac{1}{|t|} \leq 1$. Here $|t|$ denotes the order of t in $\mathbb{Z}_n$. To prove this result Lagarias and Saks needed to show that the pebbling number of an n-cube was 2^n. In 1989 Chung [1] succeeded in carrying out this idea, thus giving this alternate proof.

In this chapter we will start by describing traditional pebbling, then target pebbling, and then cover pebbling. Finally, we will describe our extension of pebbling to oriented graphs.

1.1 Traditional Pebbling

Traditional pebbling is the type of pebbling suggested by Lagarias and Saks, and used by Chung in [1]. It is usually just called pebbling, but we call it traditional pebbling to distinguish it from the other types.

In all variations of pebbling one player plays as the pebbler and the other as the configurer. For traditional pebbling the pebbler starts by buying n really expensive pebbles and gives them to the configurer. Next, the configurer distributes the n pebbles on the vertices of the graph in any way he likes and selects a target vertex t. For the rest of the game it is the pebbler's turn. The pebbler plays a sequence of pebbling moves in an attempt to place a pebble on t. The pebbler wins if he places a pebble on t. If the pebbler runs out of pebbling moves without placing a pebble on t then the configurer wins. A pebbling move is made across an edge $\{x, y\}$ by taking two pebble from x, placing one of these two pebbles on y, and removing the other pebble from play (see Figure 1.1). Moving more than one pebble across an edge can be achieved by making several pebbling moves across the edge, each move done directly after the other.

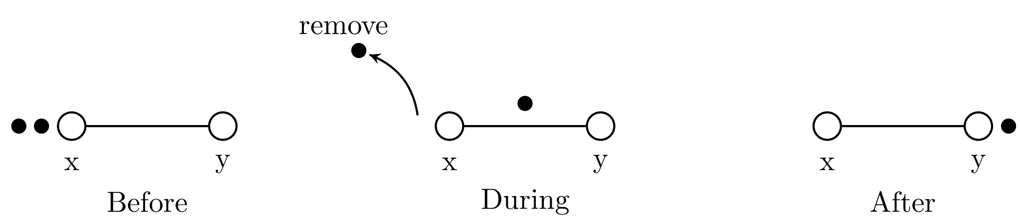


Figure 1.1: Pebbling move across an edge.

A distribution of pebbles on the vertices of a graph is called a *configuration*. A configuration C of pebbles on a graph is called t*-solvable* if starting with C the pebbler can play a sequence of pebbling moves which results in a configuration with at least one pebble on t. If this cannot be done then C is called t*-unsolvable*. If C is t-solvable for every vertex t then C is called *solvable*.

Given a graph G we would like to know the minimum number of pebbles the pebbler must buy in order to guarantee victory. This number is called the *pebbling number of* G and is denoted $\pi(G)$. For example, consider the graph G in figure 1.2.

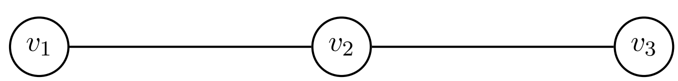


Figure 1.2

The pebbling number of G is the number $\pi(G)$ such that:

1. every configuration of $\pi(G)$ pebbles is solvable, and

2. there exists a configuration of $\pi(G)-1$ pebbles that is r-unsolvable for some vertex

r.

Let us play the game as the pebbler. Suppose we buy 3 pebbles and give them to the configurer. The configurer then places all 3 pebbles on v_1 and chooses v_3 as the target vertex (see figure below).

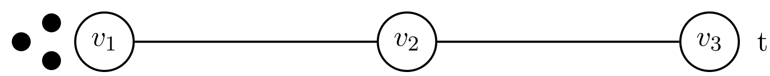


Figure 1.3

Can we place a pebble on v_3? Currently we only have one move, namely moving a pebble from v_1 to v_2. This results in the following:

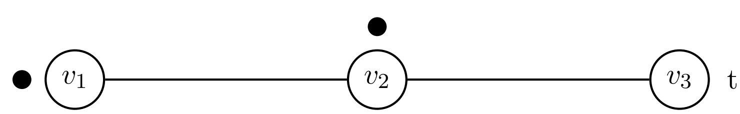


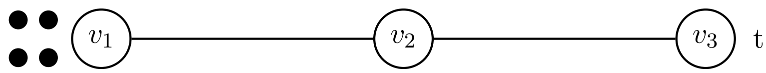
Figure 1.4

Clearly from this configuration there are no pebbling moves for us to make since no vertex has more than one pebbles on it. Since the target vertex v_3 has no pebbles on it, the pebbling number of G must be greater than 3.

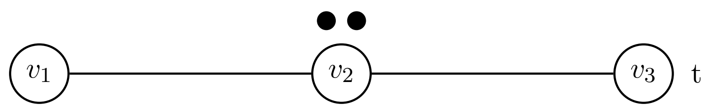
Now let us buy 4 pebbles at the beginning of the game. If the configurer places two pebbles on v_2 then no matter what target vertex he chooses we can place a pebble on it from v_2. So the pebbler must place less than two pebbles on v_2. The rest of the pebbles must be placed on the end vertices. By the pigeonhole principle one of the end vertices must have at least two pebbles on it. If v_2 is the target vertex then the pebbler can win by simply moving a pebble from the end vertex with at least two pebbles on it to v_2. Therefore the configurer must choose the target to be one of the end vertices. Without loss of generality suppose v_3 is the target vertex. Clearly v_3 must not start with any pebbles on it, otherwise we would win. This gives us only two initial configurations left to consider. The configurer can either place all pebbles on v_1, or he can place three pebbles on v_1 and one pebble on v_2.

First consider the case where all the pebbles are placed on v_1. Then we can win as follows:

Initial configuration:



Move two pebbles from v_1 to v_2 in two consecutive moves:



Move a pebble from v_2 to v_3:

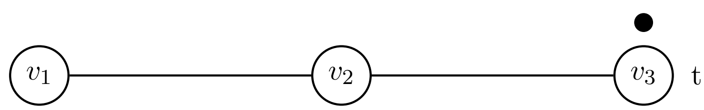
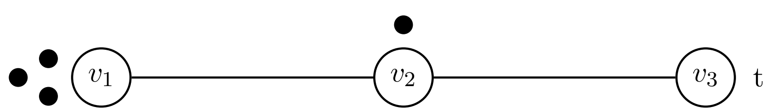


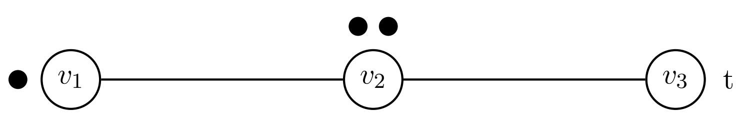
Figure 1.5

Now consider the case where three pebbles are placed on v_1 and one pebble on v_2. Then we can win as follows:

Initial configuration:



Move a pebble from v_1 to v_2:



Move a pebble from v_2 to v_3:

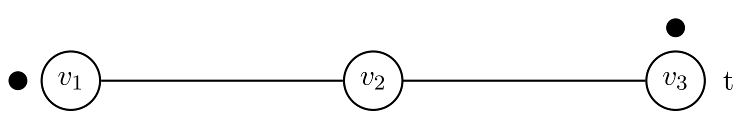


Figure 1.6

Therefore $\pi(G) = 4$. $\square$

1.2 Target Pebbling Introduced

In order to describe traditional pebbling better we will introduce another type of pebbling called target pebbling. The idea of target pebbling is not new in this paper, but our formulation of it in the light of being its own type of pebbling is. Target pebbling is often implicitly described when talking about other types of pebbling, especially traditional pebbling. It was first described (implicitly) by Chung [1] in his description of pebbling in hypercubes.

Target pebbling is a variation similar to traditional pebbling. It is played the same way as traditional pebbling except the target vertex is fixed before the game starts. Neither the pebbler nor the configurer chooses it, it is given with the graph. The game begins with the pebbler buying n really expensive pebbles and giving them to the configurer. The configurer then distributes the n pebbles on the vertices. Finally, the pebbler attempts to place a pebble on t by playing a sequence of pebbling moves. Given a graph G, the *target pebbling number of G with target t* is the minimum number of pebbles the pebbler must buy in order to guarantee he can place a pebble on t. This is denoted as $\tau(G,t)$.

Consider the graph G in figure 1.2 with target vertex $t = v_2$. Let us find $\tau(G, v_2)$ by playing as the pebbler. Suppose we buy 2 pebbles. The configurer may distribute them as follows:

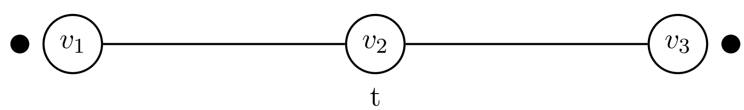


Figure 1.7

From this configuration we have no pebbling moves to make. So $\tau(G, v_2) > 2$.

Now let us start by buying 3 pebbles. If the configurer places a pebble on v_2 then we win since v_2 is our target vertex. So we only need to consider the cases where the configurer places pebbles on the end vertices. By the pigeonhole principle one of the end vertices will have at least two pebbles on it. By moving a pebble from that end vertex to v_2 we win the game. Therefore $\tau(G, v_2) = 3$.

1.3 Cover Pebbling: Another type

Cover pebbling was introduced by Crull et al. in [4]. It is a variation of traditional pebbling where instead the pebbler's goal is to place a pebble on every vertex simultaneously. The pebbler starts out by buying n really expensive pebbles and gives them to the configurer. The configurer then places the pebbles on the vertices in any way he likes. The rest of the game is played just as was done for traditional pebbling except the pebbler wins only if he reaches a distribution of pebbles where every vertex has a pebble on it at the same time. The *cover pebbling number* of a graph G, denoted $\gamma(G)$, is the minimum number of pebbles the pebbler must buy in order to guarantee victory when playing cover pebbling on G.

Consider the game of cover pebbling on the graph G in Figure 1.2. Suppose we buy 6 pebbles. The configurer then distributes the 6 pebbles as follows:

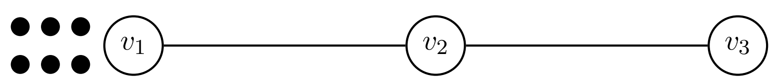


Figure 1.8

Our goal as the pebbler is to achieve a configuration with at least one pebble on every vertex. Let us start by trying to place a pebble on v_3. This can be done by moving two pebbles to v_2 and then moving one pebble from v_2 to v_3. This results in the following:

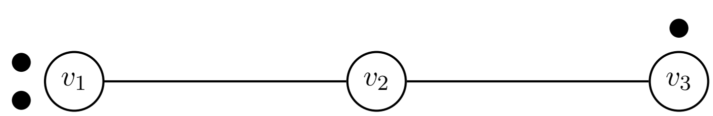


Figure 1.9

Now let us try to place a pebble on v_2. We can do this by moving a pebble from v_1 to v_2. This gives us the following:

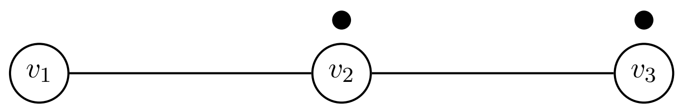


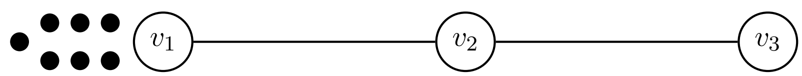
Figure 1.10

From this configuration we cannot make any more pebbling moves. Since there is no pebble on v_1 we have lost. No matter how we play we cannot win if all 6 pebbles start on v_1. So the cover pebbling number of G is greater than 6.

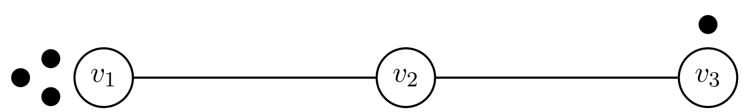
What happens if we started the game by buying 7 pebbles. If every vertex starts with a pebble on it, they we win by not making any pebbling moves. So we can assume some vertex does not have a pebble on it.

If the pebbler places all 7 pebbles on v_1 then we can win as follows:

Initial configuration:



Move two pebbles from v_1 to v_2, then move a pebble from v_2 to v_3:



Move a pebble from v_1 to v_2:

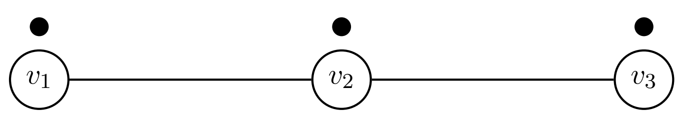


Figure 1.11

We can also win when all pebbles are placed on v_3 since this case is symmetric to when all pebbles start on v_1.

If all 7 pebbles are placed on v_2, then we can win by moving one pebble from v_2 to v_1 and one pebble from v_2 to v_3.

Now what happens if the configurer places at least one pebble on each of v_1 and v_2, but no pebbles on v_3. If v_2 has 3 or more pebbles on it, then we can win by moving a pebble from v_2 to v_3. If v_2 has at most 2 pebbles on it, then v_1 will have at least 5 pebbles on it. From here we can win by moving two pebbles from v_1 to v_2, and then move a pebble from v_2 to v_3. The case where the configurer places at least one pebble on each of v_2 and v_3 and no pebbles on v_1 is similar.

Now consider the case where v_2 is the only vertex with no pebble on it. Then either v_1 or v_3 will have at least 4 pebbles on it. WLOG suppose this vertex is v_1. Then we can win by moving a pebble from v_1 to v_2.

We have now checked all configuration of 7 pebbles on G. Since the pebbler can win from any of these configurations, and since $\gamma(G) > 6$, we must have that $\gamma(G) = 7$.

1.4 Extending Pebbling to Oriented Graphs

Pebblings on oriented graphs play the same way as on unoriented graphs except the pebbling move depends on which way we move across an edge. The *cost* to move a pebble across an edge is the number of pebbles needed to move this pebble, including the pebble itself. The *toll* is the cost when moving with the edge orientation, and the *fee* is the cost when moving against the edge orientation. The toll and fee will be at least two, and the toll will not be more than the fee.

There are many possible values we can assign to the toll and fee. We will settle for a toll of 2 and a fee of 3. In other words, to move a pebble across an edge with the orientation we must remove 1 additional pebble from the source vertex. Similarly, to move a pebble across an edge against the orientation we must remove 2 additional pebbles from the source vertex.

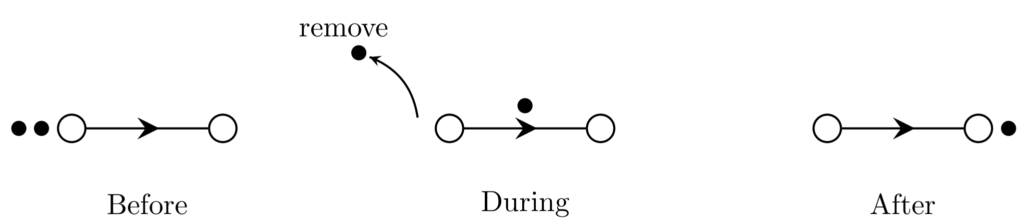


Figure 1.12: Pebbling move going with the edge orientation.

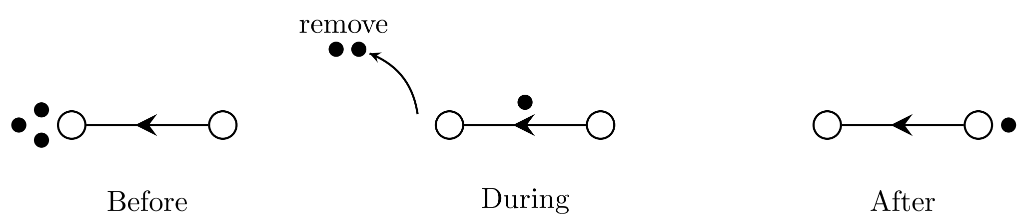


Figure 1.13: Pebbling move going against the edge orientation.

Our choice for the toll and fee may seem arbitrary at first, but our goal is to study some specific cases to make progress towards the general case which applies to any toll and fee. Often our arguments will apply easily to the general case. When it does not, we will use our choice of toll 2 and fee 3.

Let us consider an example. Suppose we have the following oriented graph G:

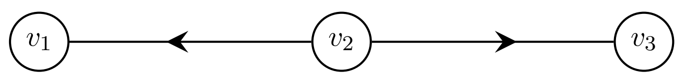


Figure 1.14

We would like to compute the traditional pebbling number $\pi(G)$. Once again, let us do this by playing as the pebbler. Suppose we buy 5 pebbles. Then the configurer chooses v_3 as the target vertex and distributes the pebbles as follows:

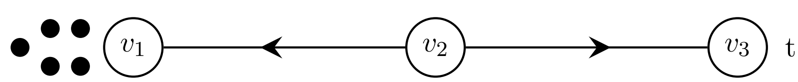


Figure 1.15

The only move we have is to move a pebble from v_1 to v_2. Since we are moving against the edge orientation we must remove 2 additional pebbles from v_1. This results in the following:

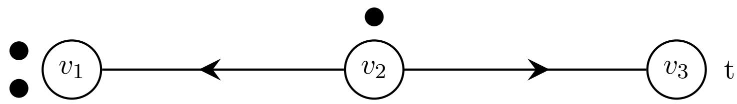
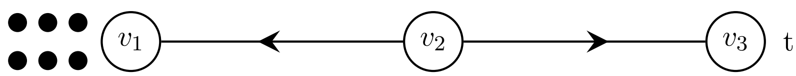


Figure 1.16

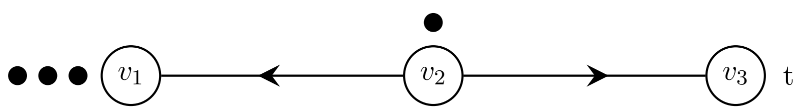
From here there are no pebbling moves to make. Since the target vertex v_3 does not have a pebble on it, $\pi(G) > 5$.

Now let us buy 6 pebbles. Suppose the configurer places all 6 pebbles on v_1 and chooses v_3 as the target. Then we can win as follows:

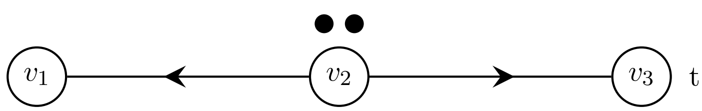
Initial configuration:



Move a pebble from v_1 to v_2:



Move another pebble from v_1 to v_2:



Move a pebble from v_2 to v_3:

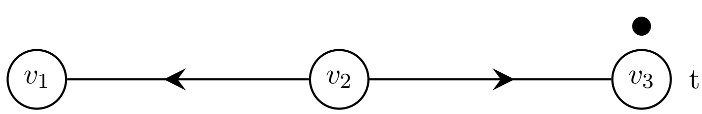


Figure 1.17

Similar reasoning as before can be used to check the rest of the configurations. So no matter how the configurer distributes the 6 pebbles on G and no matter which target vertex is chosen, the pebbler can always place a pebble on the target vertex. Therefore $\pi(G) = 6$.

Chapter 2

Definitions and Notation

This chapter is dedicated to developing the definitions and notations that we will be using. Our goal is to formalize the concepts used in pebbling of oriented graphs.

2.1 Overview of basic graph definitions

Definition 1. An *(undirected) graph* $G = (V, E)$ is an ordered pair where V is a finite set of vertices (called the vertex set of G) and E is a set of two element subsets of V (called the edge set of G). An element of V is called a *vertex* of G and an element of E is called an *edge* of G. The vertex set of G will be denoted $V(G)$ and the edge set of G will be denoted $E(G)$.

Note that the above definition does not allow edges to connect vertices to themselves.

Definition 2. Let v and u be vertices of a graph G. We say that v is *adjacent* to u if the edge $\{v, u\} \in E(G)$.

An orientation of a graphs is an assignment of a direction (called an orientation) to each edge. When drawing a graph we will indicate the orientation by drawing an arrow on the edge. So if the arrow of an edge $\{x, y\}$ points towards y then we say that this edge is oriented towards y.

Oriented graphs are graphs where each edge is assigned a direction by treating the edge as an ordered pair instead of an unordered pair. The orientation of an edge (x, y) is said to point towards y. Oriented graphs are a special case of a more general kind of graph called a directed graph (or simply digraph). We give the formal definition of a digraph below:

Definition 3. A *directed graph (digraph)* $G = (V, E)$ is an ordered pair where V is a finite set of vertices (called the vertex set of G) and E is a set of ordered pairs of V (called the (directed) edge set of G) such that if $(x, y) \in E$ then $x \neq y$. An element of V is called a *vertex* of G and an element of E is called an *edge* of G. If $(x, y) \in E$ then we say that (x, y) is oriented towards y. The vertex set of G will be denoted $V(G)$ and the edge set of G will be denoted $E(G)$.

The difference between digraphs in general and oriented graphs is that digraphs are allowed to have two edges between two vertices, one oriented in each direction. On the other hand, oriented graphs can have at most one edge between two vertices. We define oriented graphs in terms of directed graphs below:

Definition 4. An *oriented graph* G is a directed graph such that if $(x, y) \in E(G)$ then $(y, x) \notin E(G)$.

Definition 5. Let v and u be vertices of an oriented graph G. We say that v is *adjacent* to u if either $(v, u) \in E(G)$ or $(u, v) \in E(G)$.

Definition 6. The *underlying graph* of a digraph $G = (V, E)$ is the graph $G' = (V', E')$ such that

1. $V' = V$,
2. if $(x, y) \in E$ then $\{x, y\} \in E'$, and
3. if $\{x, y\} \in E'$ then either $(x, y) \in E$ or $(y, x) \in E$ or both.

Note that an oriented graph G uniquely defines an orientation of the underlying graph of G.

Definition 7. A *path* from vertex x to vertex y in a graph (or oriented graph) G is a finite sequence of distinct vertices $v_1, v_2, \ldots, v_k$ in G such that $v_1 = x$, $v_k = y$, and v_i is adjacent to v_{i+1} for all $i \in \{1, 2, \cdots, k-1\}$.

Definition 8. A vertex x is *connected* to a vertex y in a graph (or oriented graph) G if and only if there is a path from x to y in G.

Note that every vertex v in a graph G is connected to itself since the sequence (v) is a path from v to v.

Definition 9. A graph (or oriented graph) G is called *connected* if for each $x, y \in V(G)$, x is connected to y.

Definition 10. A digraph is called *(weakly) connected* if its underlying graph is connected.

Definition 11. An *oriented path graph* is an oriented graph whose underlying graph is a path graph.

The vertices along an oriented path graph will be denoted as $v_1, v_2, \ldots, v_n$ such that v_1 is adjacent to v_2, v_2 is adjacent to $v_3, \ldots, v_{n-1}$ is adjacent to v_n.

2.2 Some Pebbling Definitions

In this section we will go over some definitions and notations that apply to pebbling in general. We will also go over some miscellaneous definitions that do not fit into the other sections.

For a graph (or digraph) G we denote the set of all functions from $V(G)$ to $\mathbb{Z}$ as $\mathfrak{I}$. The set of all nonnegative functions in $\mathfrak{I}$ will be denoted as $\mathfrak{C}$.

A placement of pebbles on the vertices of a graph (or oriented graph) G is called a *configuration* on G. This can be formalized as a vertex labeling of nonnegative integers on G. A configuration can be thought of as the current state of the game. Note: some authors call configurations *distributions*. We give the formal definition below:

Definition 12. Let G be a graph (or an oriented graph). A function C is called a *configuration on* G if and only if $C \in \mathfrak{C}$.

The size of a configuration C, denoted $|C|$, is the sum $\sum\limits_{v \in V(G)} C(v)$. In other words, $|C|$ is the number of pebbles on the graph when in configuration C.

The 1-configuration C is the configuration where $C(v) = 1$ for every vertex v.

Definition 13. A configuration is called *simple* if all the pebbles are on one vertex.

Definition 14. Given a configuration C with n pebbles, to *concentrate* all pebbles from C to a vertex v means to place all pebbles on v to get the initial simple configuration with all n pebbles on v.

Definition 15. Let G be an oriented graph. Let $A = \{(x, y) : x \text{ is adjacent to } y\}$. A *cost function* $\mathbf{c}$ of G is a function from A to the set of integers greater than 1 such that for all $(a, b) \in E(G)$ and all $(a', b') \in E(G)$ we have that $\mathbf{c}(a, b) = \mathbf{c}(a', b')$ and $\mathbf{c}(b, a) = \mathbf{c}(b', a')$. In other words, the cost to move a pebble with an edge orientation is the same for all edges, and the cost to move a pebble against an edge orientation is the same for all edges. The cost to move a pebble with an edge orientation is called the *toll*, and the cost to move a pebble against an edge orientation is called the *fee*. Typically the toll is not greater than the fee.

Let's say we are moving a pebble from u to v, where u and v are adjacent vertices. The cost function represents the number of pebbles we must remove from u in order to place a pebble on v (including the pebble we moved to v). For example, if the cost from u to v is 3 (i.e., $\mathbf{c}(u, v) = 3$ where $\mathbf{c}$ is the cost function), then moving a pebble across the edge from u to v would consist of subtracting 3 pebbles from u and adding 1 pebble to v.

Definition 16. The cost function of an oriented graph will be called $\mathbf{c}$. The 2-3 *cost function* is the cost function with toll 2 and fee 3. In general, the t-f *cost function* is the cost function with toll t and fee f.

A *move* is a function from $\mathfrak{C}$ to $\mathfrak{I}$ satisfying certain conditions. A *pebbling move* is a move that maps configurations to configurations. So the image of a pebbling move is a subset of $\mathfrak{C}$.

The formal definition of a pebbling move was given in [11] as definition 2.2. We give this definition below (written using our notation):

Definition 17. Let G be a graph and let x and y be two adjacent vertices of G. For all $C \in \mathfrak{C}$, define the function $P_{x,y} : \mathfrak{C} \to \mathfrak{I}$ by

$$[P_{x,y}(C)](s) \quad = \quad \begin{cases} C(s) - 2 & \text{if } s = x \\ C(s) + 1 & \text{if } s = y \\ C(s) & \text{otherwise.} \end{cases}$$

If $P_{x,y}(C) \in \mathfrak{C}$, then P is called a *pebbling move* on C (from x to y).

For oriented graphs the number of pebbles we remove from the source vertex depends on the orientation of the edge we are crossing. Let G be an oriented graph. Suppose we are moving a pebble across an edge from x to y. If $(x, y) \in E(G)$ then we are moving with the edge orientation. So we must subtract 2 pebbles from x while adding 1 pebble to y. On the other hand, if $(y, x) \in E(G)$ then we are moving against the edge orientation. Here we must subtract 3 pebbles from x and add 1 pebble to y. In general, when moving a pebble across an edge from x to y we subtract $\mathbf{c}(x, y)$ pebbles from x and add 1 pebble to y.

We adopt the above definition of a pebbling move to oriented graphs.

Definition 18. Let x and y be two adjacent vertices of an oriented graph G. For all $C \in \mathfrak{C}$, define the function $P_{x,y} : \mathfrak{C} \to \mathfrak{I}$ by

$$[P_{x,y}(C)](s) \quad = \quad \begin{cases} C(s) - \mathbf{c}(x, y) & \text{if } s = x \\ C(s) + 1 & \text{if } s = y \\ C(s) & \text{otherwise.} \end{cases}$$

If $P_{x,y}(C) \in \mathfrak{C}$, then P is called a *pebbling move* on C (from x to y).

We often want to know weather we can obtain a configuration C' from another configuration C by a sequence of pebbling moves. We say that C' is *derivable* from C if there exists such a sequence of pebbling moves. This sequence of pebbling moves is called a *sequence of consecutive pebbling moves* from C to C'. We define both of these below by adopting Definition 2.3 given in [11].

Definition 19. Let C and C' be configurations on a graph (or oriented graph) G. Let $S = (P_1, P_2, \cdots, P_n)$ be a sequence of functions from $\mathfrak{C}$ to $\mathfrak{C}$. If P_i is a pebbling move on $P_{i-1} \circ \cdots \circ P_1(C)$ for each $i \in \{1, \cdots, n\}$, and $P_n \circ \cdots \circ P_1(C) = C'$, then we call S a *sequence of consecutive pebbling moves* from C to C'.

Definition 20. Let C and C' be configurations on a graph (or oriented graph) G. If there exists a sequence of consecutive pebbling moves from C to C' then we say that C' is *derivable* from C.

Definition 21. To *pebble* a vertex v means to make a sequence of consecutive pebbling moves to get a configuration with a pebble on v. To *pebble* a vertex v from a vertex u means to make a sequence of valid pebbling moves to move a pebble from u to v.

Definition 22. The *cost* to pebble a vertex v from a vertex u is the least number of pebbles we need to initially place on u, only placing pebbles on u, such that we can pebble v from u. The *cost* to move w pebbles from a vertex u to a vertex v is the least number of pebbles we need to initially place on u, only placing pebbles on u, such that we can move w pebbles from u to v.

For a nonnegative integer m and a vertex v we will denote the set of configurations with at least m pebbles on v as $\mathfrak{F}_m(v)$. When m is 1 we will denote this as $\mathfrak{F}(v)$.

Let ω be a configuration on a graph (or oriented graph) G. The set of configurations with at least $\omega(u)$ pebbles on every vertex $u \in V(G)$ will be denoted as $\mathfrak{W}_\omega$. When ω is the 1-configuration we will denote this as $\mathfrak{F}_\omega(v)$.

2.3 Definitions for Traditional and Target Pebbling

Since traditional and target pebbling are very similar, we will put these definitions into one section.

Definition 23. Let G be a graph (or an oriented graph) and let $t \in V(G)$. Let m be a nonnegative integer. A configuration C on G is called (t,m)-*solvable* if there exists a configuration $F \in \mathfrak{F}_m(t)$ such that F is derivable from C. C is called (t,m)-*unsolvable* if C is not (t,m)-*solvable*. When $m = 1$ we say that C is t-*solvable* or t-*unsolvable* if C is $(t,1)$-solvable or $(t,1)$-unsolvable respectively.

Definition 24. Let G be a graph (or an oriented graph) and let $t \in V(G)$. Let m be a nonnegative integer. A (t,m)-*solution* to a configuration C on G is a sequence of consecutive pebbling moves from C to a configuration in $\mathfrak{F}_m(t)$. We call a sequence of consecutive pebbling moves a t-*solution* to C if it is a $(t,1)$-solution to C.

Definition 25. Let G be a graph (or an oriented graph) and let m be a nonnegative integer. A configuration C on G is called *solvable with weight* m if for each $t \in V(G)$, C is (t, m)-solvable. C is called *unsolvable with weight* m if C is not solvable with weight m. When $m = 1$ then C is simply called *solvable* or *unsolvable* if C is solvable with weight m or unsolvable with weight m respectively.

Definition 26. Let G be a connected graph (or connected oriented graph) and let m be a nonnegative integer. The *weighted pebbling number* of G (of weight m), denoted $\pi_m(G)$, is the smallest nonnegative integer n such that every configuration on G of size n is solvable with weight m. The *pebbling number* of G, denoted $\pi(G)$, is the number $\pi_1(G)$.

Definition 27. Let G be a connected graph (or connected oriented graph). Let $t \in V(G)$, and let m be a nonnegative integer. The *target pebbling number* of G with target t and weight m, denoted $\tau_m(G, t)$, is the smallest nonnegative integer n such that every configuration on G of size n is (t, m)-solvable. The *target pebbling number* of G with target t, denoted $\tau(G, t)$, is the number $\tau_1(G, t)$.

2.4 Cover Pebbling Definitions

Definition 28. Let G be a connected graph (or connected oriented graph). Let ω be a configuration on G. A configuration C on G is called *ω-cover solvable* if there exists a configuration $W \in \mathfrak{W}_\omega$ such that W is derivable from C. C is called *ω-cover unsolvable* if C is not ω-cover solvable. When ω is the 1-configuration we say that G is *cover solvable* or *cover unsolvable* if G is ω-cover solvable or ω-cover unsolvable respectively.

Definition 29. Let G be a connected graph (or connected oriented graph). Let ω be a goal configuration on G. An *ω-cover solution* to a configuration C is a sequence of consecutive pebbling moves from C to a configuration in $\mathfrak{W}_\omega$. When ω is the 1-configuration then this is simply called a *cover solution* to C.

Definition 30. Let G be a connected graph (or connected oriented graph). Let ω be a configuration on G. The *ω-cover pebbling number* of G, denoted $\gamma_\omega(G)$, is the smallest nonnegative integer n such that every configuration of n pebbles on G is ω-cover solvable. When ω is the 1-configuration this is simply called the *cover pebbling* *number* of G and is denoted $\gamma(G)$.

Definition 31. Let G be a graph (or oriented graph). To *cover* G means to make a sequence of consecutive pebbling moves to obtain a configuration where every vertex has a pebble on it. To *cover* G from a vertex u means to cover G such that in the resulting configuration, every vertex has a pebble on it that was initially on u. To *cover* G from a set of vertices S means to cover G such that in the initial configuration every vertex in S has a pebble on it, and in the resulting configuration every vertex has a pebble on it that was initially on a vertex in S.

Definition 32. Let G be a graph (or oriented graph). The *cost* to cover G from a vertex u is the least number of pebbles we need to initially place on u, only placing pebbles on u, such that G is cover solvable from this initial configuration. The *cost* to cover G from a set of vertices S is the least number of pebbles we need to initially place on the vertices in S, placing at least one pebble on every vertex in S and no pebbles on any other vertex, such that G is cover solvable from this initial configuration.

Definition 33. Let G be a connected undirected graph. Let ω be a configuration on G. Then we define $s_\omega(G) = \max\limits_{v \in V(T)} \sum\limits_{u \in V(G)} \omega(u) \cdot 2^{d(u,v)}$, where $d(u,v)$ is the distance between u and v.

Definition 34. Let ω be a configuration on a graph G (or oriented graph). A vertex v of G is called *fat*, if the number of pebbles on it is greater than $\omega(v)$.

Chapter 3

Known Results

As stated in the introduction, graph pebbling came about because of an attempt by Lagarias and Saks to give an alternate proof to a theorem about zero-sum sequences (see introduction in [1]). This chapter will go over known results for traditional pebbling (which will include target pebbling) and for cover pebbling. In this chapter G will denote a connected graph.

3.1 Known Results for Traditional Pebbling

We will start by giving four basic facts about traditional pebbling. All of these basic facts are taken from in [6] (also reiterated in [5]). G will denote a connected graph.

Fact 1. (Depth Lower Bound, Fact 2.1 in [6]). $\pi(G) \geq 2^{diam(G)}$, where $diam(G)$ is the diameter of G.

Why is this so? If the pebbler buys less than $2^{diam(G)}$ then the configurer can choose a target vertex t and another vertex v such that the distance between t and v is $diam(G)$. Initially placing all pebbles on v results in a configuration where t cannot be pebbled.

Fact 2. (Breadth Lower Bound, Fact 2.1 in [6]). $\pi(G) \geq |V(G)|$.

If this were not the case, then the configurer can place at most one pebble on every vertex except the chosen target vertex. From here the target vertex cannot be pebbled.

Fact 3. (Pigeonhole Upper Bound, Fact 2.1 in [6]). $\pi(G) \leq (2^{diam(G)}-1)(|V(G)|-1)+1$.

Suppose this were not the case. Then by the pigeonhole principle the configurer must place at least $2^{diam(G)}$ pebbles on some vertex v. From here any target vertex can be pebbled from v.

Fact 4. (Cut Lower Bound, Fact 2.4 in [6]). If G contains a cut vertex x, then $\pi(G) > |V(G)|$.

Again, to show this assume it is not true. Let A and B be two different components of $G - x$. Let v be a vertex in A and let t be a vertex in B. Then the configurer can places 3 pebbles on v and 1 pebble on each of the other vertices except x and t. From here t cannot be pebbled.

Theorem 1. [5] $\pi(K_n) = n$.

Proof. By the breadth lower bound fact we have that $\pi(K_n) \geq n$. From the pigeonhole upper bound fact we have that $\pi(K_n) \leq (2^{diam(G)} - 1)(n-1) + 1 = (2^1 - 1)(n-1) + 1 = n - 1 + 1 = n$. $\square$

Theorem 2. [5] $\pi(P_n) = 2^{n-1}$, *where* P_n *is the undirected path graph on* n *vertices.*

Proof. We will show this by induction on n. If $n = 1$ then $\pi(P_n) = \pi(P_1) = 1 = 2^0 = 2^{n-1}$.

Now suppose $n \geq 2$. From the depth lower bound fact we have that $\pi(P_n) \geq 2^{diam(G)} = 2^{n-1}$. So we only need to show that any vertex can be pebbled from any configuration of 2^{n-1} pebbles.

Suppose 2^{n-1} pebbles are placed on P_n. Let A be the path graph from v_1 to v_{n-1} and let B be the path graph from v_2 to v_n. Then A and B are both path graphs on $n-1$ vertices. Since A and B together contain all the vertices of P_n, either $|A| \geq 2^{n-2}$ or $|B| \geq 2^{n-2}$ (recall $|G|$ is the number of pebbles on a graph G). WLOG suppose $|A| \geq 2^{n-2}$. Let t be the target vertex. If $t \in V(A)$ then by the inductive hypothesis we can pebble t. So we can assume $t \notin |A|$. In other words $t = v_n$. If $|A| \neq 2^{n-1}$ then there is a pebble on $v_n = t$. So we can assume that $|A| = 2^{n-1} = 2^{n-2} + 2^{n-2}$. By the inductive hypothesis we can place a pebble on v_{n-1} by using at most 2^{n-2} pebbles from A. After doing this A will have at least 2^{n-2} pebbles still on it. From here we can place another pebble on v_{n-1} by using the remaining 2^{n-1} pebbles on A. After placing

2 pebbles on v_{n-1} we simply make a pebbling move from v_{n-1} to v_n, pebbling the target vertex. $\square$

One of the first results in graph pebbling is the pebbling number of cubes. In 1989 Fan Chung [1] showed that the pebbling number of an n dimensional cube is 2^n.

In 1995 Pachter, Snevily, and Voxman found the pebbling number for cycles in [9]. This is given in the following theorem:

Theorem 3. [9] *For $k \geq 1$, $\pi(C_{2k}) = 2^k$ and $\pi(C_{2k+1}) = 2\lfloor 2^{k+1}/3 \rfloor + 1$.*

Pachter, Snevily, and Voxman [9] also showed that if $diam(G) = 2$ then $\pi(G) \leq |V(G)| + 1$. Since $\pi(G) \geq |V(G)|$, we get the following theorem:

Theorem 4. *If $diam(G) = 2$ then either $\pi(G) = |V(G)|$ or $\pi(G) = |V(G)| + 1$.*

We say that G is in *Class 0* if $\pi(G) = |V(G)|$. If G is not in class 0, then we say G is in *Class 1*.

In 1997 Clarke, Hochberg, and Hurlbert [2] showed that if G is 3-connected and $diam(G) = 2$ then $\pi(G) = |V(G)|$. Using this theorem they showed that almost all graphs are in class 0.

Now let us look at the Petersen graph, shown in figure 3.1.

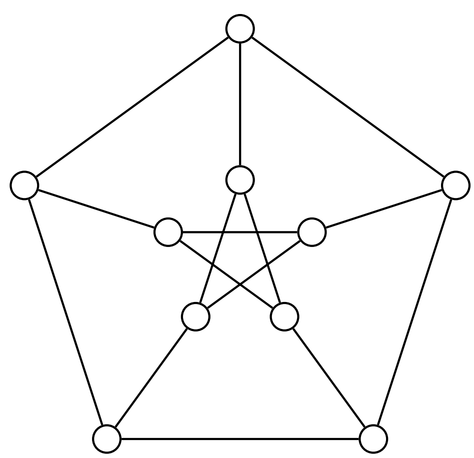


Figure 3.1: The Petersen graph.

Hurlbert [6] showed that the pebbling number of the Petersen graph is 10. One way to see this is to use the fact that the Petersen graph is 3-connected and has diameter 2. The result then follows from theorem 4.

3.2 Known Results for Cover Pebbling

Cover pebbling was introduced in 2004 by Crull et al. in their preprinted paper [3], which was published in 2005 [4]. They introduced a generalized version of cover pebbling called weighted cover pebbling. Weighted cover pebbling works the same way as normal cover pebbling except the goal configuration is specified by a weight function ω. The goal of the pebbler is to obtain a configuration where for each vertex v, v has at least $\omega(v)$ pebbles on it. In their paper [4] they established the cover pebbling number of several classes of graphs. We will list some of their results below.

Theorem 5. [4] *$\gamma(K_n) = 2n - 1$.*

Theorem 6. [4] *$\gamma_\omega(K_n) = 2|\omega| - \min\omega$, where $|\omega| = \sum_{v \in V(K_n)} \omega(v)$ and $\min\omega = \min_{v \in V(K_n)} \omega(v)$.*

Theorem 7. [4] *$\gamma(P_n) = 2^n - 1$, where P_n is the undirected path graph on n vertices.*

A *fuse* on n vertices with wick length $l-1$, denoted $F_l(n)$, is a graph on the vertices $v_1, \ldots, v_n$ such that the vertices $v_1, \ldots, v_l$ form a path and the remaining vertices are independent and adjacent only to v_l.

Theorem 8. [4] *$\gamma(F_l(n)) = (n - l + 1)2^l - 1$.*

Theorem 9. [4] *Let T be a tree. Then $\gamma_\omega(T) = s_\omega(T)$. (Recall $s_\omega(T) = \max_{v \in V(T)} \sum_{u \in V(T)} \omega(u) \cdot 2^{d(u,v)}$, where $d(u, v)$ is the distance between u and v).*

In 2006 Hurlbert and Munyan [7] showed that the cover pebbling number of an n-dimensional cube is 3^n.

3.2.1 The Stacking Theorem

By far the most important theorem for cover pebbling is the Stacking Theorem. It states that in order to find the cover pebbling number of a graph we only need to consider simple initial configurations. In other words, we can ignore any initial configuration which has pebbles on more than one vertex. This is equivalent to saying that $\gamma_\omega(G) = s_\omega(G)$.

The Stacking Theorem was proved in 2004 by Vuong and Wyckoff [11] and also independently by Jonas Sjöstrand [10]. Jonas Sjöstrand showed that for any initial configuration that is not cover solvable, all pebbles can be concentrated to one of the fat vertices and the resulting configuration will still not be cover solvable. If there are no fat vertices, than any vertex can be chosen to concentrate all the pebbles to.

Chapter 4

Traditional and Target Pebbling on Oriented Paths

In section 2.3 we defined the traditional and target pebbling numbers of a connected oriented graph G. Recall that the traditional pebbling number of G is denoted $\gamma(G)$. The target pebbling number of G with target t and weight m is denoted $\tau_m(G,t)$. When the weight is 1 we denote the target pebbling number of G as $\tau(G,t)$. Also recall that the cost function of an oriented graph is denoted $\mathbf{c}$.

Lemma 1. *Let P_n be an oriented path on n vertices. Then* $\tau_m(P_n, v_n) = m \prod_{i=1}^{n-1} \mathbf{c}(v_i, v_{i+1})$.

Proof. We will prove this by induction on n. When $n = 1$ we have $m \prod_{i=1}^{0} \mathbf{c}(v_i, v_{i+1}) = m \cdot 1 = m = \tau_m(P_1, v_1)$. Now suppose $n \geq 1$. Our inductive hypothesis says that $\tau_q(P_{n-1}, v_{n-1}) = q \prod_{i=1}^{n-2} \mathbf{c}(v_i, v_{i+1})$ (for any nonnegative integer q). So in P_{n-1} the minimum number of pebbles we need in order to guarantee that we can place $m \cdot \mathbf{c}(v_{n-1}, v_n)$ pebbles on v_{n-1} is $m \cdot \mathbf{c}(v_{n-1}, v_n) \prod_{i=1}^{n-2} \mathbf{c}(v_i, v_{i+1}) = m \prod_{i=1}^{n-1} \mathbf{c}(v_i, v_{i+1})$. After placing $m \cdot \mathbf{c}(v_n, v_{n+1})$ pebbles on v_{n-1} we simply perform m pebbling moves from v_{n-1} to v_n which result in m pebbles on v_n. We need at least $m \prod_{i=1}^{n} \mathbf{c}(v_i, v_{i+1})$ pebbles to guarantee that we can place m pebbles on v_n. Otherwise we could choose an initial configuration which cannot place $m \cdot \mathbf{c}(v_{n-1}, v_n)$ pebbles on v_{n-1}. But we need at least $m \cdot \mathbf{c}(v_{n-1}, v_n)$ pebbles on v_{n-1} to place m pebbles on v_n. Therefore $\tau_m(P_n, v_n) = m \prod_{i=1}^{n-1} \mathbf{c}(v_i, v_{i+1})$. $\square$

Remark 1. Since P_n is isomorphic to itself via the automorphism $\phi(v_i) = v_{n+1-i}$ Lemma 1 tells us that $\tau_m(P_n, v_1) = m \prod_{i=1}^{n-1} \mathbf{c}(v_{i+1}, v_i)$.

Lemma 2. *If $a, b, c, d \geq 2$ then either $a + d \leq ab$, or $a + d \leq cd$.*

Proof. Suppose $a + d > ab$. Since $b \geq 2$ we have $a + d > ab \geq 2a$. Subtracting a from both sides of this inequality yields $d > 2a - a = a$. So

$$\begin{aligned} a + d &< d + d \\ &= 2d \\ &\leq cd, \end{aligned}$$

which completes the proof. □

Theorem 10. *Let P_n be an oriented path on $n \geq 3$ vertices. If $j \in \{2, 3, \cdots, n-1\}$ then*

$$\begin{aligned} \tau(P_n, v_j) &= \prod_{i=1}^{j-1} \mathbf{c}(v_i, v_{i+1}) + \prod_{i=j}^{n-1} \mathbf{c}(v_{i+1}, v_i) - 1 \\ &< \max\{\tau(P_n, v_1), \tau(P_n, v_n)\}. \end{aligned}$$

Proof. Let A be the subgraph of P_n induced by $\{v_1, v_2, \ldots, v_j\}$. Let B be the subgraph of P_n induced by $\{v_j, v_{j+1}, \ldots, v_n\}$.

We can place $\tau(A, v_j) - 1$ pebbles on A without being able to pebble v_j. Similarly we can place $\tau(B, v_j) - 1$ pebbles on B without being able to pebble v_j. So

$$\begin{aligned} \tau(P_n, v_j) &> \tau(A, v_j) - 1 + \tau(B, v_j) - 1 \\ &= \prod_{i=1}^{j-1} \mathbf{c}(v_i, v_{i+1}) + \prod_{i=j}^{n-1} \mathbf{c}(v_{i+1}, v_i) - 2. \end{aligned}$$

Now let us start by placing $\prod_{i=1}^{j-1} \mathbf{c}(v_i, v_{i+1}) + \prod_{i=j}^{n-1} \mathbf{c}(v_{i+1}, v_i) - 1$ pebbles on P_n. If A has less than $\prod_{i=1}^{j-1} \mathbf{c}(v_i, v_{i+1})$ pebbles on it, then B has at least $\prod_{i=j}^{n-1} \mathbf{c}(v_{i+1}, v_i)$. So either A has at least $\prod_{i=1}^{j-1} \mathbf{c}(v_i, v_{i+1}) = \tau(A, v_j)$ pebbles on it or B has at least $\prod_{i=j}^{n-1} \mathbf{c}(v_{i+1}, v_i) = \tau(B, v_j)$

on it. So we can place a pebble on v_j. Therefore,

$$\tau(P_n, v_j) = \prod_{i=1}^{j-1} \mathbf{c}(v_i, v_{i+1}) + \prod_{i=j}^{n-1} \mathbf{c}(v_{i+1}, v_i) - 1.$$

Since $n \geq 3$, $2 \leq j \leq n-1$, $\mathbf{c}(v_i, v_{i+1}) \geq 2$ and $\mathbf{c}(v_{i+1}, v_i) \geq 2$ (for i from 1 to n) we have

$$\begin{aligned}
\prod_{i=1}^{j-1} \mathbf{c}(v_i, v_{i+1}) &\geq 2, \\
\prod_{i=j}^{n-1} \mathbf{c}(v_i, v_{i+1}) &\geq 2, \\
\prod_{i=1}^{j-1} \mathbf{c}(v_{i+1}, v_i) &\geq 2, \text{ and} \\
\prod_{i=j}^{n-1} \mathbf{c}(v_{i+1}, v_i) &\geq 2.
\end{aligned}$$

So by Lemma 2 either

$$\begin{aligned}
\prod_{i=1}^{j-1} \mathbf{c}(v_i, v_{i+1}) + \prod_{i=j}^{n-1} \mathbf{c}(v_{i+1}, v_i) &\leq \left(\prod_{i=1}^{j-1} \mathbf{c}(v_i, v_{i+1})\right) \cdot \left(\prod_{i=j}^{n-1} \mathbf{c}(v_i, v_{i+1})\right) \\
&= \prod_{i=1}^{n-1} \mathbf{c}(v_i, v_{i+1}) \\
&= \tau(P_n, v_n)
\end{aligned}$$

or

$$\begin{aligned}
\prod_{i=1}^{j-1} \mathbf{c}(v_i, v_{i+1}) + \prod_{i=j}^{n-1} \mathbf{c}(v_{i+1}, v_i) &\leq \left(\prod_{i=1}^{j-1} \mathbf{c}(v_{i+1}, v_i)\right) \cdot \left(\prod_{i=j}^{n-1} \mathbf{c}(v_{i+1}, v_i)\right) \\
&= \prod_{i=1}^{n-1} \mathbf{c}(v_{i+1}, v_i) \\
&= \tau(P_n, v_1).
\end{aligned}$$

So either

$$\tau(P_n, v_j) = \prod_{i=1}^{j-1} \mathbf{c}(v_i, v_{i+1}) + \prod_{i=j}^{n-1} \mathbf{c}(v_{i+1}, v_i) \, - \, 1 < \tau(P_n, v_n)$$

or

$$\tau(P_n, v_j) = \prod_{i=1}^{j-1} \mathbf{c}(v_i, v_{i+1}) + \prod_{i=j}^{n-1} \mathbf{c}(v_{i+1}, v_i) \; - \; 1 < \tau(P_n, v_1).$$

Therefore $\tau(P_n, v_j) < \max\{\tau(P_n, v_1), \tau(P_n, v_n)\}$. □

Corollary 1. *For an oriented path P_n on n vertices,* $\pi(P_n) = \max\{\prod_{i=1}^{n-1} \mathbf{c}(v_i, v_{i+1}), \prod_{i=1}^{n-1} \mathbf{c}(v_{i+1}, v_i)\}$.

Proof. If $n \leq 2$ this is trivially satisfied. So suppose $n \geq 3$. By Theorem 10 it follows that $\tau(P_n, v_j) < \max\{\tau(P_n, v_n), \tau(P_n, v_1)\}$ for all $i \in \{2, 3, \cdots, n-1\}$. So

$$\begin{aligned}
\pi(P_n) &= \max_{1 \leq i \leq n} \tau(P_n, v_i) \\
&= \max\{\tau(P_n, v_n), \tau(P_n, v_1)\} \\
&= \max\{\prod_{i=1}^{n-1} \mathbf{c}(v_i, v_{i+1}), \prod_{i=1}^{n-1} \mathbf{c}(v_{i+1}, v_i)\}.
\end{aligned}$$

□

Chapter 5

Cover Pebbling on Oriented Paths

Lemma 3. *Let P_n be an oriented path on $n \geq 3$ vertices. Let $i, j, k \in \{1, 2, \cdots, n\}$ such that $i < j < k$. Then the cost to move w pebbles from v_j to v_k is less than the cost to move w pebbles from v_i to v_k.*

Proof. The cost to move one pebble from v_j to v_k is $\prod_{t=j}^{k-1} \mathbf{c}(v_t, v_{t+1})$. Repeating this w times we get that the cost to move w pebbles from v_j to v_k is $w \prod_{t=j}^{k-1} \mathbf{c}(v_t, v_{t+1})$. Similarly, the cost to move w pebbles from v_i to v_k is $w \prod_{t=i}^{k-1} \mathbf{c}(v_t, v_{t+1})$. Since $i < j$ and $\mathbf{c}(v_t, v_{t+1}) > 0$ for all $t \in \{1, 2, \cdots, n\}$, we have that $w \prod_{t=j}^{k-1} \mathbf{c}(v_t, v_{t+1}) < w \prod_{t=i}^{k-1} \mathbf{c}(v_t, v_{t+1})$. □

Remark 2. Let i, j, and k be given as in Lemma 3. Then by symmetry the following statement follows from Lemma 3: the cost to move w pebbles from v_j to v_i is less than the cost to move w pebbles from v_k to v_i.

Lemma 4. *Let P_n be an oriented path on $n \geq 4$ vertices. Let $h, i, j, k \in \{1, 2, \cdots, n\}$ such that $h \leq i < j \leq k$. The cost to pebble v_h from v_j plus the cost to pebble v_k from v_i is more expensive than the cost to pebble v_h from v_i plus the cost to pebble v_k from v_j.*

Proof. The cost to pebble v_h from v_j is $\prod_{t=h}^{j-1} \mathbf{c}(v_{t+1}, v_t)$ and the cost to pebble v_k from v_i

is $\prod_{t=i}^{k-1} \mathbf{c}(v_t, v_{t+1})$. Similarly, the cost to pebble v_h from v_i is $\prod_{t=h}^{i-1} \mathbf{c}(v_{t+1}, v_t)$ and the cost to pebble v_k from v_j is $\prod_{t=j}^{k-1} \mathbf{c}(v_t, v_{t+1})$. Since $h \le i < j \le k$ we have $\prod_{t=h}^{i-1} \mathbf{c}(v_{t+1}, v_t) < \prod_{t=h}^{j-1} \mathbf{c}(v_{t+1}, v_t)$ and $\prod_{t=j}^{k-1} \mathbf{c}(v_t, v_{t+1}) < \prod_{t=i}^{k-1} \mathbf{c}(v_t, v_{t+1})$. So $\prod_{t=h}^{j-1} \mathbf{c}(v_{t+1}, v_t) + \prod_{t=i}^{k-1} \mathbf{c}(v_t, v_{t+1}) < \prod_{t=h}^{i-1} \mathbf{c}(v_{t+1}, v_t) + \prod_{t=j}^{k-1} \mathbf{c}(v_t, v_{t+1})$, which completes the proof. $\square$

Lemma 5. *Let P_n be an oriented path on $n > 3$ vertices. Let $h, i, j, k \in \{1, 2, \cdots, n\}$ such that $h < i < j < k$. Then the cost to pebble v_k from v_h plus the cost to pebble v_j from v_i is more expensive than the cost to pebble v_j from v_h plus the cost to pebble v_k from v_i.*

Proof. Let $a \ge 2$ be the cost to pebble v_i from v_h, $b \ge 2$ be the cost to pebble v_j from v_i, and $c \ge 2$ be the cost to pebble v_k from v_j. Then the cost to pebble v_j from v_h is ab, the cost to pebble v_k from v_i is bc, and the cost to pebble v_k from v_h is abc. So we need to show that $abc + b > ab + bc$.

Since $a \ge 2$, $b \ge 2$, and $c \ge 2$ we have that $ac + 1 > a + c$.
So $b(ac + 1) > b(a + c) \implies abc + b > ab + bc$. $\square$

Remark 3. Let h, i, j, and k be given as in Lemma 5. Then by symmetry the following statement follows from Lemma 5: the cost to pebble v_h from v_k plus the cost to pebble v_i from v_j is more expensive than the cost to pebble v_i from v_k plus the cost to pebble v_h from v_j.

Lemma 6. *Let P_n be an oriented path on $n \ge 2$ vertices. Then covering P_n only from v_1 or only from v_n is more expensive than covering P_n from both v_1 and v_n.*

Proof. Suppose P_n is being covered from v_1 and v_n. Let v_j be the vertex pebbled from v_1 that is furthest away from v_1, $1 \le j < n$. Let M be the path from v_1 to v_j and let N be the path from v_{j+1} to v_n. If $n \ge 4$ then by Lemma 3 no vertex in M is pebbled from v_n, otherwise the pebbler would use more pebbles than needed to cover P_n. If $n < 3$ then no vertex in M is pebbled from v_n since v_1 and v_j are both pebbled from v_1 and $V(M) = \{v_1, v_j\}$. So all the vertices in M are pebbled from v_1. Since v_j is the furthest vertex from v_1 being pebbled from v_1, no vertex in N is pebbled from v_1. So all the pebbles in N are pebbled from v_n.

Let A be the cost to cover M from v_1 and let B be the cost to cover N from v_n. Let $a > 0$ be the cost to pebble v_j from v_1 and let $b > 0$ be the cost to cover v_{j+1} from v_n. Let $c = \mathbf{c}(v_j, v_{j+1})$ and let $c' = \mathbf{c}(v_{j+1}, v_j)$, $c \geq 2$ and $c' \geq 2$. Note that the cost to cover v_{j+1} from v_1 is $a \cdot c$ and the cost to cover v_j from v_n is $b \cdot c'$. We will show that either the cost to cover P_n increases if we instead pebble v_j from v_n, or the cost to cover P_n increases if we instead pebble v_{j+1} from v_1. In other words, either $A - a + B + b \cdot c' > A + B$ or $A + a \cdot c + B - b > A + B$.

Suppose $A - a + B + b \cdot c' \leq A + B$. Then

$$\begin{aligned} A + B &\leq A + B + a - b \cdot c' \\ &< A + B + a \cdot c - b \cdot c' \\ &< A + a \cdot c + B - b. \end{aligned}$$

Thus one of the following will increase the cost to pebble P_n: pebble v_{j+1} from v_1 instead of from v_n, or pebble v_j from v_n instead of from v_1.

WOLG suppose pebbling v_j from v_n increases the cost. Now applying the same argument as above to this new configuration. Since the cost increased when we pebbled v_j from v_n instead of from v_1, pebbling v_{j-1} from v_n instead of from v_1 will also increase the cost. Continue this procedure until all vertices in P_n are pebbled from v_n. In each iteration we pebble v_k from v_n instead of of from v_1, where v_k is the current vertex furthest from v_1 being pebbled from v_1. By applying the above argument this will always increase the cost to pebble P_n. This procedure must stop since in each new configuration the vertex furthest from v_1 being pebbled by v_1 is closer to v_1. Therefore the cost to pebble P_n from v_n is more expensive than the cost to pebble P_n from v_1 and v_n. $\square$

Lemma 7. *Let P_n be an oriented path on $n \geq 3$ vertices. Let $j \in \{2, 3, \cdots, n-1\}$. Then covering P_n only from v_1 or only from v_n is more expensive than covering P_n from v_j.*

Proof. We will first show that either the cost to cover P_n from v_{j-1} is more expensive than the cost to cover P_n from v_j or the cost to cover P_n from v_{j+1} is more expensive than the cost to cover P_n from v_j. Let M be the path on the vertices $v_1, \ldots, v_{j-1}$ and let N be the path on the vertices $v_{j+1}, \ldots, v_n$. Let A be the cost to cover M from v_{j-1} and let B be the cost to cover B from N from v_{j+1}. Let $a = \mathbf{c}(v_{j-1}, v_j) \geq 2$, $a' = \mathbf{c}(v_j, v_{j-1}) \geq 2$, $b = \mathbf{c}(v_j, v_{j+1}) \geq 2$, $b' = \mathbf{c}(v_{j+1}, v_j) \geq 2$. Then the cost to cover P_n from v_j is $a'A+1+bB$, the cost to cover P_n from v_{j-1} is $A+a+abB$, and the cost to cover P_n from v_{j+1} is $a'b'A+b'+B$. So we must show that either $A+a+abB > a'A+1+bB$ or $a'b'A+b'+B > a'A+1+bB$.

Suppose $A+a+abB \leq a'A+1+bB$. Then

$$\begin{array}{rrcl} & A+a+abB & \leq & a'A+1+bB \\ \Longrightarrow & A+a-1+abB & \leq & a'A+bB \\ \Longrightarrow & A+a-1+abB+(a'-1)A & \leq & a'A+bB+(a'-1)A \\ \Longrightarrow & a'A+a-1+abB & \leq & (2a'-1)A+bB \\ \Longrightarrow & a'A+a-1+abB-bB & \leq & (2a'-1)A \\ \Longrightarrow & a'A+a-1+(a-1)bB & \leq & (2a'-1)A \end{array}$$

Also $a'A+a-1+(a-1)bB > a'A+1+bB$ and $(2a'-1)A < 2a'A \leq a'b'A < a'b'A+b'+B$. Therefore $a'A+1+bB < a'b'A+b'+B$. So either it is more expensive to cover P_n from v_{j-1} than from v_j or it is more expensive to cover P_n from v_{j+1} than from v_j.

WOLG suppose the cost to cover P_n from v_{j+1} is more expensive than the cost to cover P_n from v_j. Then applying the same argument as above we get that the cost to cover P_n from v_{j+2} is more expensive than the cost to cover P_n from v_{j+1}. Continue this procedure until we finally cover P_n from v_n. In each iteration we cover P_n from the next closest vertex to v_n, so this procedure must stop. Using the same argument as above we get that the cost to cover P_n in each iteration is more expensive than in the previous iteration. Therefore the cost to cover P_n from v_n is more expensive than the cost to cover P_n from v_j. □

Lemma 8. *Let P_n be an oriented path on n vertices. Let C be a most expensive cover solvable configuration on P_n. Let S be a cover solution to C. If all pebbling moves in S are made towards v_n (i.e., for each pebbling move there is a $j \in \{1, 2, \cdots, n-1\}$ such that the pebbling move is from v_j to v_{j+1}), then all pebbles in C are on v_1.*

Proof. Suppose for a contradiction that not all pebbles in C are on v_1. Then there is a pebble p on a vertex v_i for some $i \in \{2, 3, \cdots, n\}$. By Lemma 3 placing p on v_1

instead of on v_i will increase the cost to pebble v_i, and thus increase the cost to cover P_n. So C is not the most expensive cover solvable configuration. But we assumed it was. Contradiction. □

Remark 4. Let C and S be given as in Lemma 8. Then by symmetry the following statement follows from Lemma 8: if all pebbling moves in S are made towards v_1, then all pebbles in C are on v_n.

Lemma 9. *Let P_n be an oriented path on $n \geq 3$ vertices. Let C be a most expensive cover solvable configuration on P_n. Let S be a cover solution to C that uses the least number of pebbles (i.e., one that removes the least number of pebbles from play). If there is a pebbling move in S from v_j to v_{j+1} for some $j \in \{1, 2, \cdots, n-2\}$, then there is no pebbling move in S from v_{j+2} to v_{j+1}.*

Proof. Suppose for a contradiction that a pebbling move is made from v_{j+2} to v_{j+1}. Since there is a pebbling move from v_j to v_{j+1}, by Lemma 4 there is no pebbling move from v_{j+1} to v_j. Similarly, there is no pebbling move from v_{j+1} to v_{j+2}. So in the final configuration of S there is at least two pebbles on v_{j+1}. But then the pebbler could have used less pebbles to cover P_n by not making the pebbling move from v_{j+2} to v_{j+1}. So S does not use the least number of pebbles to cover P_n. But we assumed S did. Contradiction. □

Corollary 2. *Let C and S be given as in Lemma 9. If $j \leq n-2$ and there is a pebbling move in S from v_j to v_{j+1}, then either there is a pebbling move from v_i to v_{i+1} for each $i \in \{j, j+1, \cdots, n-1\}$, or there is an $m \in \{j+1, j+2, \cdots, n-1\}$ such that there is no pebbling move in either direction across the edge $\{v_m, v_{m+1}\}$.*

Proof. Suppose there is an $i \in \{j+1, j+2, \cdots, n-1\}$ such that there is no pebbling move in S from v_i to v_{i+1}. Then there is a smallest such i, call it m. Since m is smallest, there is a pebbling move from v_{m-1} to v_m. So by Lemma 9 there is no pebbling move in S from v_{m+1} to v_m. Therefore there is no pebbling move in any direction across the edge $\{v_m, v_{m+1}\}$. □

Remark 5. By symmetry we can reverse the direction of pebbling in Lemma 9 and Corollary 2, making pebbling moves towards v_1 instead of towards v_n.

Theorem 11. *(Stacking Theorem for oriented paths) Let P be an oriented path. To determine $\gamma(P)$ it is sufficient to consider simple initial configurations. In fact, for any initial configuration of P that is not cover solvable, there is an end vertex such that all pebbles can be concentrated to this end vertex and the resulting configuration will still not be cover solvable.*

Proof. Suppose the stacking theorem for oriented paths is not true. Then there is a counter example with the smallest number of vertices, say n. In other words, for all orientations of any path P_m where $m < n$ the theorem holds.

Case 1. There is a $j \in \{2, 3, \cdots, n-1\}$ such that v_{j-1} is pebbled from v_j and v_{j+1} is pebbled from v_j. Let M be the path from v_1 to v_j and let N be the path from v_j to v_n. Since M and N are both shorter than P_n, they are not counter examples. Pebblings of M or N are mutually independent by Lemma 4. Hence, the most expensive pebbling of P_n is a combination of most expensive pebblings of M and N, respectively. By our assumption, the most expensive pebbling of M is from a single end vertex — either from v_1 or from v_n. Because there are pebbles on v_j used to pebble M, it must be v_j. Similarly, the most expensive pebbling of N is from v_j. By Lemma 7 it is more expensive to cover P_n from one of its end vertices than it is to cover P_n from v_j. This means our counter example is not most expensive. Contradiction.

Case 2. There is no v_j as in Case 1 and there is no edge with no pebbling move across it. Then by Corollary 2 all edges have pebbling moves in one direction, say towards v_n. Now apply Lemma 8 to get a contradiction.

Case 3. There is no v_j as in Case 1 and for some $i \in \{1, 2, \cdots, n-1\}$ there is an edge $\{v_i, v_{i+1}\}$ with no pebbling moves made across it in either direction. Let M be the path from v_1 to v_i and let N be the path (possibly trivial) from v_{i+1} to v_n. Again, pebblings of M and N are mutually independent, and by our assumption each of these is pebbled from one of its end vertices. We are left to consider three cases (up to symmetry).

Subcase 3.1. We only have pebbles on v_1 and v_n. Then by Lemma 6 this is not the most expeisive configuration. Contradiction.

Subcase 3.2. We only have pebbles on v_1 and v_{i+1}. If $i = n-1$ then $i+1 = n$ and this is treated by Subcase 3.1. If $i < n-1$ then obviously covering N from v_1 is more expensive.

Subcase 3.3. We only have pebbles on v_i and v_{i+1}. If $i < n-1$ then we can amalgamate v_i and v_{i+1} together, treating them as one vertex. Then we are looking at P_{n-1} with vertices $v_1, v_2, \cdots, v_{i-1}, v_i = v_{i+1}, v_{i+2}, \cdots, v_n$. In this new path, all pebbles are on the single vertex $v_i = v_{i+1}$. By Lemma 7 this is not the most expensive configuration for P_{n-1}, so the original configuration was not most expensive. Contradiction.

If $i = n-1$ then we have pebles only on v_{n-1} and v_n. But pebbling P_n only from v_n is more expensive. Contradiction. □

Corollary 3. *Let P_n be an oriented path. Then*

$$\gamma(P_n) = \max\{\sum_{i=1}^{n} \prod_{j=1}^{i-1} \mathbf{c}(v_j, v_{j+1}), \sum_{i=1}^{n} \prod_{j=i}^{n-1} \mathbf{c}(v_{j+1}, v_j)\}.$$

Proof. By Theorem 11 to compute $\gamma(P_n)$ we only need to compute two costs: the cost to cover P_n from v_1 and the cost to cover P_n from v_n. The cost to cover P_n is the maximum of these two values.

To cover P_n from v_1 we first pebble v_n, then v_{n-1}, and so on until we finally pebble v_1. Each time we make a pebbling move we are careful to only move pebbles towards v_n. In this way we minimize the number of pebbles needed to cover P_n. This gives us a total cost of $\sum_{i=1}^{n} \prod_{j=1}^{i-1} \mathbf{c}(v_j, v_{j+1})$.

The process to cover P_n from v_n is done similarly. We first pebble v_1, then v_2, and so on until we finally pebble v_n. This gives us a total cost of $\sum_{i=1}^{n} \prod_{j=i}^{n-1} \mathbf{c}(v_{j+1}, v_j)$. Therefore

$$\gamma(P_n) = \max\{\sum_{i=1}^{n} \prod_{j=1}^{i-1} \mathbf{c}(v_j, v_{j+1}), \sum_{i=1}^{n} \prod_{j=i}^{n-1} \mathbf{c}(v_{j+1}, v_j)\}.$$

□

Chapter 6

Conclusion

In this paper we only looked at a special case of the Stacking Theorem, namely the Stacking Theorem for oriented paths. We conjecture that the Stacking Theorem is true for any oriented graph. Perhaps approaches taken in [11] and [10] could be used. We state our conjecture below.

Conjecture 1. (Stacking Conjecture) Let ω be a positive configuration of an oriented graph G. To determine $\gamma_\omega(G)$ it is sufficient to consider simple initial configurations.

There are many more questions one could ask about pebbling done on oriented graphs. For paths it would be interesting to see how changing the edge orientation affects the pebbling number. A good starting point would be to see what effect swapping orientations between edges has. Here, the number of edges oriented in each direction does not change, just their positions change.

Another good direction would be to look at the pebbling number of other oriented graphs. What is the pebbling and cover pebbling number of oriented cycles? How do these values compare to their unoriented versions? It would be natural to split this into two cases: even cycles and odd cycles.

What is the pebbling and cover pebbling number of oriented starts? How about oriented fuses? What about oriented trees? My guess would be that oriented trees would be similar to the unoriented version and use the stacking "theorem."

We hope this paper will spark interest in pebbling of oriented graphs, and in general to the development of pebbling of directed graphs. There is a lot more waiting to be explored; some surprises waiting around the corner.

References

[1] F. R. K. Chung. Pebbling in hypercubes. *SIAM J. Disc. Math*, 2:467–472, 1989.

[2] T. Clarke, R. Hochberg, and G. Hurlbert. Pebbling in diameter two graphs and products of paths. *J. Graph Theory*, 25:119–128, 1997.

[3] B. Crull, T. Cundiff, P. Feltman, G. Hurlbert, L. Pudwell, Z. Szaniszlo, and Z. Tuza. The cover pebbling number of graphs, 2004. preprint.

[4] B. Crull, T. Cundiff, P. Feltman, G. Hurlbert, L. Pudwell, Z. Szaniszlo, and Z. Tuza. The cover pebbling number of graphs. *Discrete Math*, 296:15–23, 2005.

[5] G. Hurlbert. Hurlbert's pebbling page: Pebbling numbers, Accessed August 2015. http://www.people.vcu.edu/ ghurlbert/pebbling/pnum.html.

[6] G. Hurlbert. A survey of graph pebbling. *Congressum Numerantium*, 139:41–64, 1999.

[7] G. Hurlbert and B. Munyan. Cover pebbling hypercubes. *Bull. Inst. Combin. Appl.*, 47:71–76, 2006.

[8] P. Lemke and D. Kleitman. An addition theorem on the integers modulo n. *J. Number Theory*, 31:335–345, 1989.

[9] L. Pachter, H. Snevily, and B. Voxman. On pebbling graphs. *Congressum Numerantium*, 107:65–80, 1995.

[10] J. Sjöstrand. The cover pebbling theorem. *Electronic Journal of Combinatorics*, 12:N22, 2005.

[11] A. Vuong and M. I. Wyckoff. Conditions for weighted cover pebbling of graphs, 18 October 2004. http://arXiv.org/abs/math.CO/0410410.